Express Mail Label No. EL930548002US

[0001] # INSERTION SORTER

BACKGROUND

[0003] The invention generally relates to wireless communication systems. In particular, the invention relates to an insertion sorter used in conjunction with estimation of noise variance within a time division multiple access (TDMA), or time division-code division multiple access (TD-CDMA) receiver.

[0004] Communication systems using TDMA and TD-CDMA signals are well known in the art. For simplicity, both types of signals are referred to as TDMA hereinafter. In communications between a user equipment (UE) and a base station, the TDMA signal has a frame structure with a plurality of time slots. For certain types of signals, each of the time slots includes two data bursts that are separated by a user specific midamble. The data bursts transmit the desired data and, the user specific midamble is used to perform channel estimation. The midamble contains a series of chips, which in turn are processed through a filter to produce a series of channel response taps.

[0005] Among the series of channel response taps are signal taps, which represent the communication signal, with the remaining taps representing noise. Depending on the specific type of TDMA system architecture, the number of signal taps is designated by a fixed predetermined value.

[0006] A TDMA receiver must screen the channel response taps to determine which taps are the signal taps. The signal taps are those taps having the greatest value. A sorter is conventionally used to identify the most significant taps as the signal taps. Several iterations of the sorter are typical of a system using conventional bubble sort methods. It is desirable to provide a sorter with a minimal number of hardware components and arranged to provide a

high degree of sorting efficiency.

[0007] SUMMARY

[0008] The present invention provides a system having a sorter circuit which determines a selected number of greatest values from a set of values. The system is used for a TDMA receiver as an insertion sorter for identifying peak values of a communication channel response and determining the sum of the non-peak values. The resultant values are then used in a conventional manner to process received communication data.

[0009] For a sort depth N, the sorter circuit is configured to store N peak channel response values sorted in descending order from a set of L values. The remaining L-N channel response values are considered to be noise and are summed using a single adder, and stored in a single register as an overall noise value. The sorter circuit comprises N series–connected sorter elements. Each sorter element has a comparator and a register. A set of channel response values is sequentially processed by inputting each value simultaneously to all the sorter elements in parallel. Processing the set of channel responses with parallel inputs to each sorter element minimizes the operating time of the system such that the number of clock cycles is equal to the number of channel response values processed.

[0010] Noise variance of the communication signal can be calculated by applying to the sum of the non-peak values a predetermined scaling factor appropriate for the specific type of communication system.

[0011] Other objects and advantages of the invention will be apparent to those skilled in the art from the following description.

[0012] BRIEF DESCRIPTION OF THE DRAWINGS

[0013] FIG.1 shows a block diagram for a single sorter element of an insertion sorter circuit.

[0014] FIG. 2 shows a block diagram for an insertion sorter circuit with series-connected multiple sorter elements.

[0015] FIG. 3A shows a typical TDMA midamble with 57 channel response taps.

[0016] FIG. 3B shows a table of insertion sorter register contents for several clock pulses.

[0017] FIG. 4 shows a block diagram of an insertion sorter circuit with three sorter elements.

[0018] FIG. 5 shows the algorithm for using the insertion sorter to determine noise variance of a communication signal.

[0019] FIG. 6 shows a block diagram of the channel response square accumulator that derives the input for the insertion sorter circuit of FIGs. 2 and 4.

[0020] DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0021] The present invention is described with reference to the drawing figures where like numerals represent like elements throughout. Referring to FIG. 1, there is shown a sort element 150, which comprises the basic building block of a sorter circuit 200 shown in FIG. 2. Preferably, the sorter circuit 200 is used to sort sets of communication signal channel response power values (CR values), but can be utilized to sort any set of random values.

[0022] FIG. 1 illustrates the configuration of a sorter element 150, which is adapted for downstream connection to a like sorter element. The sorter element 150 comprises a multiplexer (MUX) switch 100_i, a register 101_i, a comparator 102_i and an AND gate 104_i.

[0023] The sorter element 150 has two outputs, namely an enable output 120_i from the comparator 102_i and a shift output 130_i from the register 101_i. The sorter element 150 has two inputs, namely an enable input 120_{i-1} for the AND gate 104_i and a shift input value 130_{i-1} for the MUX 100_i. As shown in FIG. 2, the sorter element 150 is connected downstream of a like circuit by coupling the downstream shift input to the upstream shift output, such that both are represented by 130_i.

[0024] The sorter element 150 also has a value input CR associated as input to both the MUX 100_i and the comparator 102_i. When multiple sorter elements 150 are serially connected in a cascade for processing a set of CR values, individual CR values are input in parallel to all of the sorter elements for processing. The parallel input of the CR values for processing results in a processing cycle time for each CR value to be equal to the processing cycle time for one sorter element, since the same CR value is processed by each of the sorter elements during the same cycle.

[0025] The CR value input is an "A" input to the comparator 102_i. The comparator also has a "B" input 130_i that receives the current value of register 101_i. If the CR value is greater than the current register value, i.e. the "A" input is greater than the "B" input, the enable output 120_i of comparator 102_i is a "high" value, otherwise it is a "low" value.

[0026] The MUX 100_i is coupled to the register 101_i by a MUX output 125_i to output either the CR value from the CR input or the upstream register's value via the shift input 130_{i-1}. The AND gate 104_i is coupled to the MUX 100_i via an AND gate output 121_i. Output 121_i of AND gate 104_i is high when both enable inputs 120_i and 120_{i-1} are high, otherwise the output 121_i is low. When output 121_i is high, the MUX 100_i directs the value input 130_{i-1} to register 101_i; when output 121_i is low, the MUX 100_i directs the CR input to register 100_i via MUX output 125_i.

[0027] Register 101_i receives a clock pulse during each cycle, which triggers the register to change its value by loading the output 125_i of the MUX 100 if the comparator output 120_i is high. Otherwise the register value remains unchanged.

[0028] Where sorter element 150 is connected in a series of like elements, when comparator output 120_i is high, the output of all downstream comparators are also high. This results in both inputs of each AND gate 104_i of all downstream sorter elements to also be high, so that the value of register 100_i is passed to the register of the next downstream element in each case. Thus, starting with the first register that has a value lower than the CR

value being processed, the register values shift downwardly while maintaining an automatic sort of the processed values.

[0029] As shown in FIG. 2, any desired number N of sorter elements $150_1 \ldots 150_N$ may be serially connected to configure a sorter circuit 200. The first sorter element 150_1 is modified slightly to eliminate the MUX and the AND gate since the only input to the first sorter circuit 150_1 is the CR value input.

[0030] The last sorter element 150_N in sorter circuit 200 has its comparator's enable output 120_N and register's value output 130_N coupled to a MUX 110. MUX 110 has a CR value input and outputs the CR value to a summer/accumulator 105 during each cycle unless the comparator enable output 120_N is high. For a high enable output 120_N, output 130_N from register 101_N is passed through MUX 110 to summer/accumulator 105. Summer/accumulator 105 adds the value input from MUX 110 to a stored value. That sum is output to a register 106 as a noise value and is also returned to the summer as its stored value for the next cycle.

[0031] In operation, sorter circuit 200 receives a new CR value for each cycle. Upon the clock pulse being sent to all the registers in the sorter elements 150_1-150_N, that CR value will either be stored in one of the registers or passed through MUX 110 to summer 105. Where the CR value is stored in one of the registers 150_1-150_N, all downstream comparator outputs will be high so that the value of the register in sorter element 150_N will pass through MUX 110 to be summed by the summer/accumulator 105. Accordingly, at the completion of processing of an arbitrary number L of random values, the N highest values will be stored in registers 101_1-101_N in descending order and all values not so stored will have been summed by summer/accumulator 105. Where the set of random values are CR values which include N signal values, the N signal values will be contained in registers 150_1-150_N and the remaining values will have been summed to represent a noise value in register 106.

[0032] Although the sorter circuit 200 is particularly useful in sorting CR values, it can be

employed to sort any set of values. If the set has L values and L < N, all L values will reside in the registers 150_1-150_N and the summer/accumulator 105 output will be 0.

[0033] FIG. 2 shows insertion sorting circuit 200, which has a desired number N sort elements $150 \ldots 150_N$ cascaded to produce a sort depth N. Although the limit to the number of N sort elements is a factor of desired circuit size and power consumption considerations, time limits are not constraining because of the parallel input characteristics of the series of N sorter elements. During each process cycle, register 101_i of each sorter element 150 in sorting circuit 200 keeps its currently stored value, or updates its value with the current channel response CR value, or updates its value with the stored channel response CR value of the register 101_{i-1} directly upstream.

[0034] Turning to FIG. 3A, a data stream representative of a TDMA midamble of length L_m=57 chips is shown. These are processed to provide a set of 57 channel response taps from which the CR values are derived for processing in accordance with the sorter circuit of the present invention. FIG. 6 shows a block diagram of a circuit for producing CR values. Channel response taps are comprised of real and imaginary components. The real channel response taps are stored by registers R_{CRR1} and R_{CRR2} as duplicate values, and then squared by a multiplier 201. Similarly, the imaginary channel response taps are stored in duplicate in registers R_{CRI1} and R_{CRI2} and squared by a multiplier 202. An adder 203 receives the squared real and imaginary values and sends the sum to a register PSA. Register PSA accordingly stores real values since the square of the imaginary components results in a real number. These values are the preferred CR values that are sorted by the insertion sorter circuit 200.

[0035] A predetermined number N of CR values are intended to represent an estimate of channel responses that contain the actual communication signal, while the remaining number M of channel responses have values that are less than each of the N peak values, and thus are presumed to be noise on the channel. Accordingly, for the example shown in FIG. 3A, the

number M of channel responses that represent noise are: $M = L_m - N = 57 - 3 = 54$.

[0036] It should be recognized that the number L_m of channel responses may be a number other than 57, and that the number N of signal elements can also vary. Some values typical in TDMA systems are $L_m = 28, 32, 64, 57$ and 114, while signal element values are typically N=6 or N=10. In the following example, an arrangement of sorter elements where N=3 will be described for simplicity. FIG. 4 shows an insertion sorter circuit 300 with three registers 100_1, 100_2, 100_3 that contain sorter element values P1, P2, and P3, respectively. At time t=0, each sorter element is initialized so that registers 101_1, 101_2, and 101_3 contain the value 0. This is reflected in the register table shown in FIG. 3B at time t=0. As shown in FIG. 3A, it is assumed that a sequence of 57 channel responses will be loaded into the insertion sorter circuit 300, one value at a time, with each clock pulse.

[0037] At time t=1, the first channel response value CR=10 is present at the A-side of comparators 102_1, 102_2 and 102_3, the input of register 101_1, and the low (0) input of multiplexers 100_2 and 100_3. Simultaneously, each comparator 102_1, 102_2, 102_3 evaluates the expression $10 > 0$ and produces a high output to the enable signals 120_1, 120_2 and 120_3. Register 101_1 receives the high enable signal 120_1 and accordingly loads the value "10" from its input. MUX 100_2 receives a high enable input 121_2 from AND gate 104_2, while register 101_2, with its high enable input 120_2, loads the value "0" from register 101_1.

[0038] Similarly, the value "0" from register 101_2 is transferred through MUX 100_3 and subsequently loads into register 101_3, since MUX 100_3 and register 101_3 have a high enable signal 121_3. The initial content value P3=0 for register 101_3 is passed through MUX 110 to the input of adder 105 and on to register 106 as the first stored NOISE value. Accordingly, as shown in FIG. 3B at time t=1, the first clock pulse produces register values of P1=10, P2=0, P3=0 and NOISE=0.

[0039] In the next clock pulse at t=2, the channel response value CR=3 is loaded into each sort element of sorter circuit 300 at the low (0) input of multiplexers 100_2, 100_3, and 110,

and the "A" inputs of comparators 102_1, 102_2 and 102_3. The channel response value CR=3 is present at register 101_1, but register 101_1 does not load this value since condition (A < B) is present at comparator 102_1 and the enable signal 120_1 is low as a result. Register 101_2 loads the value "3" from the low side of MUX 100_2, as enable signal 121_2 is low at MUX 100_2 and enable signal 120_2 is high at register 101_2. The previously stored value P2=0 from register 101_2 is passed through MUX 100_3 and loaded into register 101_3, since enable signals 120_3 and 121_3 are high. MUX 110 receives the value "0" from register 101_3, which in turn is loaded into input "A" of adder 105, and is summed with the sum NOISE=0 stored from the prior clock pulse t=0. The new total value NOISE at register 106 becomes: NOISE = A+B = 0+0=0. Consequently, the sort element values after the second clock pulse are P1=10, P2=3, P3=0, and NOISE = 0, as shown in FIG. 3B at t=2.

[0040] The foregoing process is repeated at each successive clock pulse. After the occurrence of the third clock pulse (t=3), the third channel response CR=9 is retained by the second register 101_2 and the second channel response value CR=3 is shifted to the third register 101_3. Not until the channel response CR=N+1, does the insertion sorter circuit 300 produce a noise value, since there are N registers representative of the channel signal. Accordingly, after the fourth clock pulse (t=4), the insertion sorter circuit 300 recognizes the lowest of the first four channel response CR values as NOISE, which in this example is the fourth channel response CR=2. At the fifth clock pulse (t=5), the channel response value CR=12 is loaded into the first register 101_1, replacing the prior peak value P_1 from the fourth clock pulse, P_1=10. Register 101_2 receives the value "10" from MUX 100_2, and the former content of register 101_2 is stored into register 101_3 via MUX 100_3. The value "3" from register 101_3 is passed through MUX 110, sent to adder 105 port "A", and is added to the value NOISE=2 for a total noise value NOISE=5 at register 106.

[0041] This process is repeated until all 57 CR values are sorted by sorter insertion circuit 300. In the end, register 101_1 will contain the peak channel response value, $P1_{MAX}$, register

101_2 will contain $P2_{MAX}$ and register 101_3 will contain $P3_{MAX}$ where $P1_{MAX} \geq P2_{MAX} \geq P3_{MAX}$. After 57 clock pulses, register 106 will contain the total noise value NOISE, which is the sum of M= 54 noise values.

[0042] FIG. 5 shows an algorithm that makes use of the insertion sorter circuit 200. Input signal CR represents the squared sum of the real and imaginary channel response taps produced by circuit 400 of FIG. 6. The insertion sorter circuit 200 embodies the block 510, 515 and 525 functions. The block 510 of selection and sorting of the N most significant channel response elements is achieved by the comparisons performed by the sorter element comparators $102_1 \ldots 102_N$, and the storage and shifting of the sorter element registers $101_1 \ldots 101_N$. The sorter element comparators $102_1 \ldots 102_N$ and registers $101_1 \ldots 101_N$ likewise perform the block 515 function of selection of the M least significant channel response elements by passing and not storing the least significant values in registers $101_1 \ldots 101_N$. The block 525 sum function is performed by the adder 105 and is stored in register 106. Block 520 is implemented by tapping of the registers $101_1 \ldots 101_N$ and summing the sorted values stored by them once the processing of the series of CR values is completed (not shown in FIG. 2) to form value P_G, which represents the channel power estimate. The noise variance σ^2 is produced by multiplying the value P_S, the sum of the noise values stored in register NOISE, with a predetermined scaling factor C_S, which is a function of the number of paths and the channel length for the particular system.

[0043] Although the invention has been described in part by making detailed reference to certain specific embodiments, such detail is intended to be instructive rather than restrictive. It will be appreciated by those skilled in the art that many variations may be made in the structure and mode of operation without departing from the spirit and scope of the invention as disclosed in the teachings herein.

* * *